

CAMERA-TRAP IMAGES SEGMENTATION USING MULTI-LAYER ROBUST PRINCIPAL COMPONENT ANALYSIS

Jhony-Heriberto Giraldo-Zuluaga^{}, Alexander Gomez^{*}, Augusto Salazar^{*}, and Angélica Diaz-Pulido[†]*

^{*} Grupo de Investigación SISTEMIC, Facultad de Ingeniería, Universidad de Antioquia UdeA,
Calle 70 No. 52-21, Medellín, Colombia

[†] Instituto de Investigación de Recursos Biológicos Alexander von Humboldt,
Calle 28A No. 15-09, Bogotá D.C, Colombia

ABSTRACT

Camera trapping is a technique to study wildlife using automatic triggered cameras. However, camera trapping collects a lot of false positives (images without animals), which must be segmented before the classification step. This paper presents a Multi-Layer Robust Principal Component Analysis (RPCA) for camera-trap images segmentation. Our Multi-Layer RPCA uses histogram equalization and Gaussian filter as pre-processing, texture and color descriptors as features, and morphological filters with active contour as post-processing. The experiments focus on computing the sparse and low-rank matrices with different amounts of camera-trap images. We tested the Multi-Layer RPCA in our camera-trap database. To our best knowledge, this paper is the first work proposing Multi-Layer RPCA and using it for camera-trap images segmentation.

Index Terms— Camera-trap images, Multi-Layer Robust Principal Component Analysis, background subtraction, image segmentation.

1. INTRODUCTION

Studying and monitoring of mammals and birds species can be performed using non-invasive sampling techniques. These techniques allow us to observe animal species for conservation purposes, e.g. to estimate population sizes of endangered species. Camera trapping is a method to digitally capture wildlife images. This method facilitates the register of terrestrial vertebral species, e.g. cryptic species. Consequently, camera traps can generate large volumes of information in short periods of time. Thus, the contributions in camera trapping are important for better species conservation decisions.

Camera traps are devices to capture animal images in the wild. These devices consist of a digital camera and a motion detector. They are triggered when the motion sensor detects movement and dependent on the temperature of the source in relation to the environment temperature. Biologist can monitor wildlife with camera traps for detecting rare species, delinquent species distributions, monitoring animal behavior, and

measuring other biological rates [1]. Camera traps generate large volumes of information, for example a camera trapping study can generate until 200000 images, where 1% of the information is valuable [2]. As a consequence, biologists have to analyze thousands of photographs in a manual way. Nowadays, software solutions cannot handle the increment of the number of images in camera trapping [3]. Accordingly, it is important to develop algorithms to assist the post-processing of camera-trap images.

Background subtraction techniques could help to segment animals from camera-trap images. There is a significant body research of background subtraction focused in video surveillance [4]. Nevertheless, there are not enough methods that can handle the complexity of natural dynamic scenes [5]. Camera-trap images segmentation is important for animal detection and classification. Camera-traps images usually have ripping water, moving shadows, swaying trees and leaves, sun spots, scene changes, among others. Consequently, the models used to segment those types of images should have robust feature extractors. There are some segmentation methods in the literature applied to camera-trap images segmentation. Reddy and Aravind proposed a method to segment tigers on camera-trap images, using texture and color features with active contours [6]. They do not make an objective evaluation of their method. Ren et al. developed a method to segment images from dynamic scenes, including camera-trap images; the method uses Bag of Words (BoW), Histogram of Oriented Gradients (HOG), and graph cut energy minimization [7]. They do not show the results on camera-trap images. Zhang et al. developed a method to segment animals from video sequences, using camera-trap images; the method uses BoW, HOG, and graph cut energy minimization [8]. They obtained 0.8695 of average f-measure on their own camera-trap data set.

Robust Principal Component Analysis (RPCA) is a method derived from Principal Component Analysis. RPCA assumes that a data matrix can be decomposed in a low-rank and a sparse matrix. RPCA has newly seen significant activity in many areas of computer sciences, particularly in

background subtraction. As a result, there are some algorithms to solve the RPCA problem [9, 10, 11, 12, 13]. In this work, we proposed a Multi-Layer RPCA approach in order to segment animals from camera-trap images. Our method combines color and texture descriptors as feature extractor, and solve the RPCA problem with some state-of-the-art algorithms. To our knowledge, this paper is the first work in proposing a Multi-Layer RPCA approach and using it for camera-trap images segmentation.

The paper is organized as follows. Section 2 shows material and methods. Section 3 describes the experimental framework. Section 4 presents the experimental results and the discussion. Finally, Section 5 shows the conclusions.

2. MATERIALS AND METHODS

This section shows a brief explanation of the algorithms and metrics used in this paper.

2.1. Robust Principal Component Analysis

An image can be decomposed in a low-rank and sparse matrix. Equation 1 shows the RPCA problem, where M is the data matrix, L_0 is the low-rank matrix, and S_0 is the sparse matrix. The low-rank matrix is the background and the sparse matrix is the foreground in background subtraction.

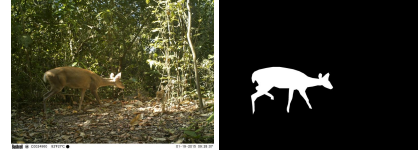
$$M = L_0 + S_0 \quad (1)$$

The RPCA problem can be solved with the convex program Principal Component Pursuit (PCP). This program computes L and S , taking the objective function in the Equation 2, where $\|L\|_*$ denotes the nuclear norm of the low-rank matrix, $\|S\|_1$ denotes the l_1 -norm of the sparse matrix, and λ is a regularizing parameter. There are some algorithms to perform PCP such as Accelerated Proximal Gradient (APG), and Augmented Lagrange multiplier (ALM) [14].

$$\begin{aligned} &\text{minimize} \quad \|L\|_* + \lambda\|S\|_1 \\ &\text{subject to} \quad L + S = M \end{aligned} \quad (2)$$

2.2. Multi-Layer Robust Principal Component Analysis

Equation 3 shows the data matrix M in our Multi-Layer RPCA method, where $\beta \in [0, 1]$ is a weight value indicating the contribution of the texture function to the overall data matrix. Function $f_t(x, y)$ denotes the texture descriptor extracted from each image, using the classic Local Binary Pattern (LBP) [15]. LBP describes the texture of an image using the neighborhood of each pixel. Function $f_c(x, y)$ denotes the color transformation for each image, converting to gray scale in this case. Our Multi-Layer RPCA computes the L and S matrices from the M matrix in the Equation 3. Texture descriptors can work robustly into rich texture regions with light variation. However, they do not work in a efficient



(a) Original Image (b) Ground truth

Fig. 1: Ground truth example for the evaluation process, (a) original image, (b) ground truth or manual segmented image.

way on uniform regions such as water, the sky, and others. Color descriptors could overcome the texture descriptor limitation [16]. The Multi-Layer approach proposed in this paper was tested in camera traps for wildlife image segmentation.

$$M(x, y) = \beta f_t(x, y) + (1 - \beta) f_c(x, y) \quad (3)$$

2.3. Evaluation Metrics

The f-measure metric was chosen to evaluate the performance of the Multi-Layer RPCA. Equation 4 shows the f-measure, where precision and recall are extracted from the confusion matrix. Precision is the proportion of predicted positives that are correctly real positives. In the same way, recall denotes the proportion of the real positives that are correctly predicted [17]. The confusion matrix is computed comparing the ground truth (GT) with the automatic segmented images.

$$\text{f-measure} = 2 \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (4)$$

3. EXPERIMENTAL FRAMEWORK

This section introduces the database used in this paper, the experiments executed, and the implementation details of our Multi-Layer RPCA.

3.1. Database

The Alexander von Humboldt Institute realizes samplings with camera traps in different regions of the Colombian forest. We select 25 cameras from 8 regions, where each camera has a relative unalterable environment. Each camera was placed in its site between 1 to 3 months. We extract 30 days and 30 nights of images from those cameras, in daytime color and nighttime infrared formats respectively. The database consists of 1065 GT images from the 30 days and 30 nights. The images have a spatial resolution of 3264x2448 pixels. The length of each day or night data set varies from 9 to 72 images, depending on the animal activity that day or night. Figure 1 shows an example of the GT images.

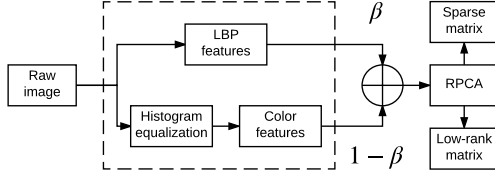


Fig. 2: Block diagram of the pre-processing methods used in the Experiment 1, 3, and 4.

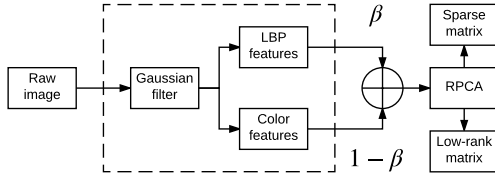


Fig. 3: Block diagram of the pre-processing methods used in the Experiment 2 and 3.

3.2. Experiments

The experiments computed the background models with different conditions and amount of images, observing the robustness of the Multi-Layer RPCA and the influence of pre-processing in the results. All experiments performed our method with $\beta = [0, 0.05, 0.1, 0.15, \dots, 1]$. Accordingly, Experiment 1 uses histogram equalization as pre-processing in the color transformed image, Figure 2 shows the pre-processing for each raw image. The background model is computed with entire days e.g. we take all images of day 1 and solve the RPCA problem in the Experiment 1. Experiment 2 computes the background model with entire nights; Figure 3 shows the pre-processing for each raw image. Experiment 3 takes entire days and nights e.g. we take all images of day 1 and night 1 to solve the RPCA problem. Experiment 3 uses two pre-processes, daytime images uses the pre-process in Figure 2 and nighttime images uses the pre-process in Figure 3. Experiment 4 takes entire days and nights such as the Experiment 3, but it only uses the pre-processing in Figure 2 for all images.

We tested 9 algorithms to solve the RPCA problem in this paper. Active Subspace RPCA (AS-RPCA) [9]; Exact ALM (EALM), Inexact ALM (IALM), Partial APG (APG-PARTIAL) and APG [10]; Lagrangian Alternating Direction Method (LSADM) [11]; Non-Smooth Augmented Lagrangian v1 (NSA1) and Non-Smooth Augmented Lagrangian v2 (NSA2) [12]; Probabilistic Robust Matrix Factorization (PRMF) [13].

The foreground was obtained applying a post-process to the sparse matrix. The post-processing was the same for all experiments. This stage includes a hard threshold, morphological filters, and an active contours with a negative contraction bias [18]. Figure 4 shows the post-processing used. Fi-

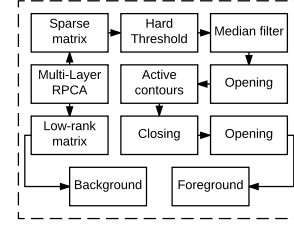


Fig. 4: Block diagram of the postprocessing.

nally, The f-measure was computed comparing each GT with each foreground. The average f-measure was computed as the mean of all f-measures. The results are displayed as a plot of the average f-measure vs β .

3.3. Implementation Details

The RPCA algorithms were computed using the Sobral et al. library [19]. The rest of the source code was developed using the image processing toolbox of Matlab.

4. RESULTS

This section shows the results and discussions of the experiments introduced in the Section 3. These results use the metrics explained in the Section 2.3.

Figures 5a and 5b show the average f-measure vs β of the Experiments 1 and 2 for all RPCA algorithms chosen. Table 1 shows the summary of the best results for each experiment. APG-PARTIAL was the best algorithm in the Experiments 1 and 2. Daytime images have rich texture regions. In contrast, nighttime images have uniform color. Texture representations are more important on daytime images in the Experiment 1 due to $\beta = 0.6$. On the contrary, color descriptors are more important on nighttime images in the Experiment 2 due to $\beta = 0.3$. Those results show the importance of combining the color and texture descriptors. Figure 5 shows the performances of the RPCA normal algorithms when $\beta = 0$. Thus, our Multi-Layer RPCA outperforms the RPCA normal methods.

Figures 5c and 5d show the average f-measure vs β of the Experiments 3 and 4. Table 1 shows that dividing the pre-processing per daytime or nighttime in the Experiment 3 does not make a big difference in the results, but it increases the fine-tuning parameters. Table 1 shows that NSA2 was the best algorithm in the Experiments 3 and 4, contrary to the Experiments 1 and 2 where APG-PARTIAL was the best. NSA2 algorithm is a better choice than other RPCA algorithms, if we cannot differentiate between daytime and nighttime images, or if it is difficult to do so. On the other hand, APG-PARTIAL is better, if we have information about the infrared activation.

Figure 6 shows two visual results of the Multi-Layer

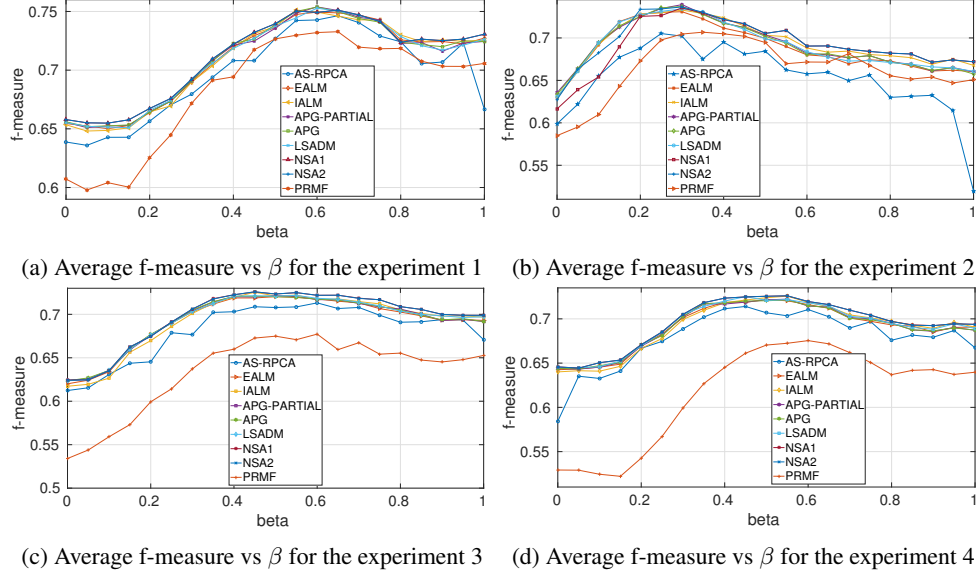


Fig. 5: Results of the proposed experiments, (a) average f-measure vs β per days, (b) average f-measure vs β per nights, (c) average f-measure vs β per days and nights with two different pre-processes, (d) average f-measure vs β per days and nights with one pre-process.

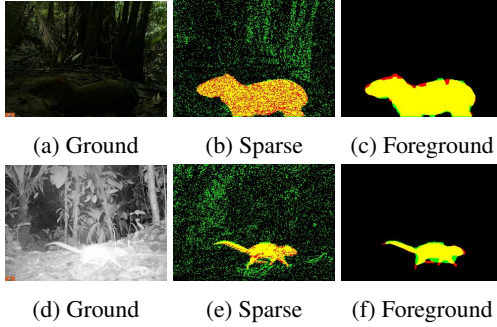


Fig. 6: Visual results of the Multi-Layer RPCA, (a) original daytime image, (b) sparse matrix after the hard threshold with APG-PARTIAL and $\beta = 0.6$, (c) foreground, (d) original nighttime image, (e) sparse matrix after the hard threshold with APG-PARTIAL and $\beta = 0.3$, (f) foreground.

RPCA. Figure 6a shows a daytime image without any pre-processing. Figure 6d shows an original nighttime image. Figures 6b and 6e show the sparse matrix after the hard threshold. Figures 6c and 6f show the foreground image. These color results are made with the GT images. Yellow-colored regions mean pixels that are on the GT and the automatic segmented images. Red and green regions are visual representations of the under and over segmentation.

5. CONCLUSIONS

We proposed a Multi-Layer RPCA for camera-trap image segmentation, using texture and color descriptors. The pro-

Table 1: β values and algorithms for the best performances of each experiment.

Experiment	Algorithm	β	Avg f-measure
Experiment 1	APG-PARTIAL	0.6	0.7539
Experiment 2	APG-PARTIAL	0.3	0.7393
Experiment 3	NSA2	0.45	0.7259
Experiment 4	NSA2	0.55	0.7261

posed algorithm is composed of pre-processing, RPCA algorithm, and post-processing. The pre-processing uses histogram equalization, Gaussian filtering, or a combination of both. The RPCA algorithm computes the sparse and low-rank matrices for background subtraction. The post-processing computes morphological filters and an active contours with a negative contraction bias. We proved the Multi-Layer RPCA algorithm in a camera-trap images database from the Colombian forest. The database was manually segmented to extract the f-measure of each automatic segmented image. We reach 0.7539 and 0.7393 of average f-measure in daytime and nighttime images respectively. The average f-measure was computed with all GT images. To our best knowledge, this paper is the first work in proposing Multi-Layer RPCA and using it for camera-trap images segmentation.

Acknowledgment. This work was supported by the Colombian National Fund for Science, Technology and Innovation, Francisco José de Caldas - COLCIENCIAS (Colombia). Project No. 111571451061.

6. REFERENCES

- [1] A. F. O’Connell, J. D. Nichols, and K. U. Karanth, *Camera traps in animal ecology: methods and analyses*. Springer Science & Business Media, 2010.
- [2] A. Diaz-Pulido and E. Payan, “Densidad de ocelotes (*leopardus pardalis*) en los llanos colombianos,” *Mastozoología neotropical*, vol. 18, no. 1, pp. 63–71, 2011.
- [3] E. H. Fegraus, K. Lin, J. A. Ahumada, C. Baru, S. Chandra, and C. Youn, “Data acquisition and management software for camera trap data: A case study from the team network,” *Ecological Informatics*, vol. 6, no. 6, pp. 345–353, 2011.
- [4] S. Brutzer, B. Höferlin, and G. Heidemann, “Evaluation of background subtraction techniques for video surveillance,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1937–1944, IEEE, 2011.
- [5] V. Mahadevan and N. Vasconcelos, “Spatiotemporal saliency in dynamic scenes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 171–177, 2010.
- [6] K. P. K. Reddy and R. Aravind, “Segmentation of camera-trap tiger images based on texture and color features,” in *Communications (NCC), 2012 National Conference on*, pp. 1–5, IEEE, 2012.
- [7] X. Ren, T. X. Han, and Z. He, “Ensemble video object cut in highly dynamic scenes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1947–1954, 2013.
- [8] Z. Zhang, T. X. Han, and Z. He, “Coupled ensemble graph cuts and object verification for animal segmentation from highly cluttered videos,” in *Image Processing (ICIP), 2015 IEEE International Conference on*, pp. 2830–2834, IEEE, 2015.
- [9] G. Liu and S. Yan, “Active subspace: Toward scalable low-rank learning,” *Neural computation*, vol. 24, no. 12, pp. 3371–3394, 2012.
- [10] Z. Lin, M. Chen, and Y. Ma, “The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices,” *arXiv preprint arXiv:1009.5055*, 2010.
- [11] D. Goldfarb, S. Ma, and K. Scheinberg, “Fast alternating linearization methods for minimizing the sum of two convex functions,” *Mathematical Programming*, vol. 141, no. 1-2, pp. 349–382, 2013.
- [12] N. S. Aybat, D. Goldfarb, and G. Iyengar, “Fast first-order methods for stable principal component pursuit,” *arXiv preprint arXiv:1105.2126*, 2011.
- [13] N. Wang, T. Yao, J. Wang, and D.-Y. Yeung, “A probabilistic approach to robust matrix factorization,” in *European Conference on Computer Vision*, pp. 126–139, Springer, 2012.
- [14] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?,” *Journal of the ACM (JACM)*, vol. 58, no. 3, p. 11, 2011.
- [15] T. Ojala, M. Pietikainen, and T. Maenpää, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [16] J. Yao and J.-M. Odobez, “Multi-layer background subtraction based on color and texture,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2007.
- [17] D. M. Powers, “Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation,” 2011.
- [18] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic active contours,” *International journal of computer vision*, vol. 22, no. 1, pp. 61–79, 1997.
- [19] A. Sobral, T. Bouwmans, and E.-h. Zahzah, “Lrslibrary: Low-rank and sparse tools for background modeling and subtraction in videos,” in *Robust Low-Rank and Sparse Matrix Decomposition: Applications in Image and Video Processing*, CRC Press, Taylor and Francis Group, 2015.